FEBRUARY 1994 ISSUE # 44

HUNTER VALLEY VZ JOURNAL

HV VZ USERS GROUP
※ NEW ADDRESS ※

35 TIGHES TERRACE
TIGHES HILL
NSW 2297

PHONE - 049 611 474

PRODUCED BY HV VZ USER'S GROUP
A NON PROFIT PUBLICATION
EDITOR: JOSEPH P LEON

FRONT COVER

IT SAYS IT ALL. I HAVE MOVED AGAIN, BUT NOT VERY FAR AND HAVE NO PLANS MOVING AGAIN UNLESS I WIN LOTTO OR SOMETHING SIMILIAR. IT IS VERY NICE HAVING A PLACE OF MY OWN AGAIN AFTER BOARDING FOR A COUPLE YEARS WITH MY BROTHER.

*** HELP / SELL & TELL ***

APOLOGIES . . .

MY SINCEREST APOLOGIES TO ALL VZ JOURNAL READERS FOR THE VERY, VERY LATENESS OF THIS ISSUE. AS MENTIONED IN LAST ISSUE I HAVE A MEDICAL PROBLEM WHICH GOT MUCH WORSE NEAR COMPLETION OF MY NEW HOUSE. IT AFFECTED ALL MY JOINTS AND MADE LIFE UNBEARABLE FOR MANY MONTHS.

I HAD TO BE OUT OF MY LAST ADDRESS BY A CERTAIN DATE AND SO I PUSHED MYSELF AND AM STILL PAYING FOR IT NOW. I HAD MANY RELAPSES SINCE THEN BUT THANKFULLY I'M STARTING TO FEEL NEAR NORMAL. THE NEXT ISSUE SHOULD BE ON TIME AS I'VE LEARNED MY LESSON AND AM TAKING THINGS EASY.

NEXT/FUTURE ISSUE/S . . .

FINAL PART OF VZ FREQUENCY.
MULTIPLE USR CALL ROUTINE (USED IN VZ FREQUENCY).
MACHINE CODE DIRECTORY ROUTINE (USED BY VZ FREQUENCY).
MINI AMPLIFIER FOR THE VZ.
MANUAL DATA INPUT FOR VZ LOTTO.
JOURNAL INDEX PART II

FOR SALE - FOR SALE . . .

1 OFF VZ 300
2 OFF DISK DRIVES
1 OFF PRINTER INTERFACE
1 OFF GREEN SCREEN MONITOR
1 OFF LIGHT PEN
1 OFF DR20 TAPE DATA RECORDER
1 OFF 16 MEMORY EXPANSION
1 OFF DISK CONTROLLER
1 OFF JOYSTICK INTERFACE
1 OFF 8 BIT PORT (HV VZ J #30)
1 OFF MODEM AND INTERFACE
ASSORTED GAME CASSETTES
BOX OF ASSORTED DISKS, BOOKS, MAGAZINES AND MANUALS

$250.00 ONO. WILL SEPERATE BUT WOULD PREFER TO SELL AS ONE LOT.

ALSO FOR SALE SPARE PARTS INCLUDING:

APPROXIMATELY 6 VZ'S, EXPANSION CASES, MODEMS, JOYSTICKS, MEMORY, CPU CHIPS (SOME NEW), MONITORS (GOING).

BEST OFFER FOR ABOVE . . .

BEN HOBSON PO BOX 255 QUIRINDI 2343 (067) 462 076 AFTER 6.30 PM

PUBLIC DOMAIN SOFTWARE

IT HAS BEEN SUGGESTED TO ME THAT I ASSEMBLE A NUMBER OF DISKS CONTAINING PUBLISHED PROGRAMS, PUBLIC DOMAIN SOFTWARE, ETC AND OFFER TO READERS AT MINIMAL COST TO COVER EXPENSES. PLEASE LET ME KNOW YOUR IDEAS ON THIS AND WHAT TYPE OF SOFTWARE YOU WOULD LIKE. ED.

BACK ISSUES . . .

MOST BACK ISSUES ARE IN STOCK AND AVAILABLE AT FOLLOWING PRICES.

CURRENT ISSUE	- $3.50 EACH	- INCLUDES POSTAGE
1 - 5 ISSUES	- $3.00 EACH	- INCLUDES POSTAGE
6 OR MORE ISSUES	- $2.75 EACH	- INCLUDES POSTAGE
OUT OF STOCK ISSUES	- $3.50 EACH	- INCLUDES POSTAGE

LIST IS A SMALL ROUTINE WHICH LISTS A BASIC PROGRAM STORED ON DISK TO THE PRINTER (OR SCREEN AFTER A SMALL MOD).

SO WHATS NEW? I HEAR YOU ASK.

WELL, THE BEAUTY OF THIS UTILITY IS THAT IT CAN BE CALLED FROM A BASIC PROGRAM AND DOES NOT RE-ENTER IMMEDIATE MODE AFTER LISTING THE PROGRAM, (I.E. THE CALLING BASIC PROGRAM KEEPS RUNNING).

WHAT'S THE USE?

THE MAIN USE OF LIST IS TO PROVIDE AN EASY WAY TO LIST MANY BASIC PROGRAMS IN A "BATCH LIKE" PROCESS WITHOUT REQUIRING INTERVENTION BY THE USER.

HOW TO USE LIST.

(1) A CONTROLLING BASIC PROGRAM IS REQUIRED. IT MUST LOAD THE ROUTINE, SET UP THE USR ADDRESS AND PROVIDE THE FILENAMES OF THE PROGRAMS TO LIST.

(2) THE CONTROLLING BASIC PROGRAM MUST BE RELOCATED SUCH THAT ITS START ADDRESS IS ABOVE THE END ADDRESS OF ALL OF THE BASIC PROGRAMS IT IS LISTING.

(3) X=USR("FILENAME") IS THE SYNTAX REQUIRED TO LIST A FILE. AFTER THE CALL, X INDICATES WHETHER AN ERROR OCCURRED (NON-ZERO = ERROR).

RELOCATING A BASIC PROGRAM

THE EASIEST WAY TO RELOCATE A PROGRAM IS TO USE THE EXT12 UTILITY. AN ALTERNATIVE IS TO MOVE THE START OF BASIC POINTER AS DESCRIBED IN THE TECHNICAL MANUAL.

RUNNING A RELOCATED BASIC PROGRAM

IF YOU TRY TO RUN A RELOCATED BASIC PROGRAM AND YOU GET A SYNTAX ERROR TRY THE FOLLOWING:

```
SOB=(PEEK(30884)+256*PEEK(30885))-1
POKE SOB,0 OR POKE SOB-65536,0 IF SOB > 32767
```

NOW TYPE RUN

HERE IS AN EXAMPLE OF A CONTROLLING BASIC PROGRAM (NOTE THAT LIST.BIN IS LOCATED AT 29184)

```
10 BLOAD"LIST.BIN"
20 POKE30862,0:POKE30863,114
30 POKE29282,0 'FOR SCREEN LIST
40 X=USR("FILE1")
50 X=USR("FILE2")
60 X=USR("FILE3")
70 X=USR("ETC  ")
80 X=USR("ETC  ")
```

LIST SOURCE CODE ON NEXT 2 PAGES

```
;***************************
;*  MACHINE CODE ROUTINE   *
;* PRINTS A BASIC PROGRAM *
;***************************
;
;NOT RELOCATABLE CODE.
;
;***************************
;ROM SUBROUTINES.
;
PCHR EQU  032AH
PINT EQU  0FAFH
FLNE EQU  1B2CH
PCR  EQU  20FEH
IMOD EQU  2828H
PMSG EQU  2B75H
DTOK EQU  2B7EH
PWON EQU  4008H
PWOF EQU  400BH
LOAD EQU  4041H
;
;***************************
;BASIC PROGRAM POINTERS
;
DVCE EQU  789CH
SBAS EQU  78A4H
BPTR EQU  78A7H
RTYP EQU  78AFH
EBAS EQU  78F9H
ARG  EQU  7921H
;
;***************************
;CONSTANT DEFINITIONS
;
VINT EQU  2
VSTR EQU  3
SDEV EQU  0
PDEV EQU  1
FTYP EQU  "T"
STAK EQU  7700H
;
;***************************
;
;CHECK ARG IS A STRING
     CP   VSTR
     JP   NZ,ERR
;CHECK IF IMMEDIATE MODE
     CALL IMOD
;SAVE DOS VECTOR
     LD   (DOS),IY
;CLEAR FILENAME BUFFER
     LD   DE,(DOS)
     LD   HL,(DOS)
     INC  HL
     INC  DE
     INC  DE
     LD   (HL),32
     LD   BC,7
     LDIR
;GET STRING ARG (FILENAME)
     LD   HL,(ARG)
;A=STRING LENGTH
     LD   A,(HL)
     INC  HL
;IF STRING LEN=0, RETURN
     CP   0
     JR   Z,ERR
;ENSURE STRING LEN <= 8
     CP   9
     JP   M,LP1
     LD   A,8
LP1  LD   E,(HL)
     INC  HL
     LD   D,(HL)
     EX   DE,HL
;COPY FILENAME TO BUFFER
     LD   DE,(DOS)
     INC  DE
     LD   C,A
     LD   B,0
     LDIR
;SAVE STACK POINTER
     LD   (STCK),SP
;SET STACK PTR TO FREE MEM
     LD   SP,STAK
;SAVE IMPORTANT BASIC PTRS
     LD   HL,(SBAS)
     PUSH HL
     LD   HL,(EBAS)
     PUSH HL
     LD   HL,(BPTR)
     PUSH HL
;SET FILETYPE TO BASIC TYPE
     LD   (IY+9),FTYP
;LOAD FILE SPECIFIED FROM
;DISK IN CURRENT DRIVE
     CALL PWON
     CALL LOAD
;SAVE ERROR CODE.
     PUSH AF
     CALL PWOF
;RESTORE THE ERROR CODE
     POP  AF
;WAS THERE A DISK ERROR?
     CP   0
     JR   NZ,ERR2
;SET THE OUTPUT DEVICE TO
;THE PRINTER
     LD   A,PDEV
     LD   (DVCE),A
;SET THE BASIC INPUT BUFFER
;TO OUR OWN BUFFER
     LD   HL,IBUF
     LD   (BPTR),HL
;SEARCH FOR THE FIRST BASIC
;PROGRAM LINE
     LD   DE,0
     CALL FLNE
;SAVE FOUND LINE PTR
     PUSH BC
LOOP POP  HL
;BC = PTR TO NEXT LINE
     LD   C,(HL)
     INC  HL
```

```
        LD    B,(HL)
        INC   HL
;CHECK IF END OF PROGRAM
        LD    A,B
        OR    C
        JR    Z,LEXX
;SAVE PTR TO NEXT LINE
        PUSH BC
;BC = LINE NUMBER
        LD    C,(HL)
        INC   HL
        LD    B,(HL)
;SAVE LINE NUMBER
        PUSH BC
;HL = TOKENISED LINE PTR
        INC   HL
;SAVE LINE PTR. HL=LINE NO
        EX    (SP),HL
;PRINT THE LINE NUMBER
        CALL PINT
;OUTPUT A SPACE
        LD    A,32
        CALL PCHR
        POP   HL
;DETOKENISE STRING POINTED
;TO BY HL
        CALL DTOK
;LIST LINE ON OUTPUT DEVICE
        LD    HL,IBUF
        CALL PMSG
;OUTPUT A CR
        CALL PCR
        JR    LOOP
;
;EXIT FROM ROUTINE ON ERROR
ERR     LD    HL,1
        JR    LEX1
ERR2 EQU      $
;
;RESTORE IMPORTANT BASIC
;PROGRAM POINTERS
        POP   HL
        LD    (BPTR),HL
        POP   HL
        LD    (EBAS),HL
        POP   HL
        LD    (SBAS),HL
        LD    HL,1
        JR    LEX1
;
;EXIT FROM ROUTINE NO ERROR
LEXX EQU  $
;
;OUTPUT TWO CR'S
        CALL PCR
        CALL PCR
;RESTORE IMPORTANT BASIC
;PROGRAM POINTERS
        POP   HL
        LD    (BPTR),HL
        POP   HL
        LD    (EBAS),HL
        POP   HL
        LD    (SBAS),HL
;RESET OUTPUT DEVICE TO
;THE SCREEN
        LD    A,SDEV
        LD    (DVCE),A
;HL=RET VALUE. ZERO=SUCCESS
        LD    HL,0
LEX1 LD       (ARG),HL
;SET RETURN ARG TYPE TO INT
        LD    A,VINT
        LD    (RTYP),A
;RESTORE STACK POINTER
        LD    SP,(STCK)
        RET
;
;***************************
;DATA STORAGE AREA
;
IBUF DEFS 100
STCK DEFW 0
DOS  DEFW 0
```

DOS BUGS REPORT 2 IN (DSE) DOS V1.2
BY LESLIE MILBURN

I HAVE FOUND YET ANOTHER BUG IN DOS V1.2. THIS TIME IT IS TO DO WITH THE DCOPY COMMAND. IF DCOPYING A SINGLE FILE USING TWO DRIVES AND THE DESTINATION DISK IS WRITE PROTECTED, THE DESTINATION DRIVE IS LEFT SPINNING AND NO ERROR MESSAGE IS DISPLAYED.

IF ANYBODY HAS FOUND ANY OTHER BUGS IN ANY DOS SYSTEM, PLEASE INFORM THE EDITOR WHO CAN PASS THE INFORMATION ON TO OTHER USERS.

NOTE: IF WRITING TO EDITOR PLEASE NOTE NEW ADDRESS AND PHONE NUMBER WHICH IS SHOWN ON FRONT COVER.

44-7 BASIC COMMAND ABBREVIATIONS
BY BOB KITCH

A NUMBER OF ABBREVIATED FORMS OF BASIC COMMANDS WILL RUN ON THE VZ. THESE SHORTENED FORMS ARE NOT FULLY DOCUMENTED IN THE MANUALS ACCOMPANYING THE MACHINES; SO I HAVE SET THEM OUT BELOW. TRY THEM, AS THEY SAVE A LITTLE SPACE ON THE 32 CHARACTER LONG LINE. THEY SHOULD BE USED WITH CAUTION AS THEY CONTRIBUTE LITTLE TO PROGRAM READABILITY.

FULL COMMAND	ABBREVIATED COMMAND
10 REM A NON-EXECUTABLE LINE	10 ' A NON-EXEC....
20 PRINT"SEE THIS""	20 ?"SEE THIS"
30 SOUND 1,3:SOUND 2,5	30 SOUND 1,3;2,5
40 LET A=23	40 A=23
50 IF A=1 THEN GOTO 999	50 IF A=1,GOTO 999 ...OR... 50 IF A=1,999
60 PRINT"ENTER VALUE":INPUT A	60 INPUT"ENTER VALUE";A

TO PUT THE INPUT "?" ANY WHERE ON THE SCREEN USE:

```
70 PRINT@488;:INPUT A
```

MULTIPLE NEXT STATEMENTS ARE NOT REQUIRED IF THEY FOLLOW EACH OTHER IN THE PROGRAM.

EDITOR'S NOTE:

THE USE OF (') INSTEAD OF REM SAVES TWO CHARACTER SPACES ON THE PROGRAM LINE BUT USES 3 BYTES OF MEMORY INSTEAD OF 1 BYTE.

THE USE OF (?) INSTEAD OF PRINT SAVES CHARACTER SPACES ON THE PROGRAM LINE, BUT ONCE ENTERED THE VZ CONVERTS IT TO THE PRINT STATEMENT. EACH NUMBERED LINE ON THE VZ CONSISTS OF 64 CHARACTER SPACES INCLUDING THE LINE NUMBER.

IT IS POSSIBLE BY USING (?) MANY TIMES ON A LINE TO FORCE THE VZ TO EXCEED THE 64 CHARACTER LIMIT. THE VZ WILL DISPLAY IT, BUT WILL NOT ALLOW EDITING OF THE EXTRA CHARACTERS/LINES AND IF YOU PRESS RETURN ON THE LINE THE EXTRA CHARACTERS/LINES WILL DISAPEAR.

WHEN RENUMBERING YOUR PROGRAM THE USE OF (') INSTEAD OF REM CAN CORRUPT YOUR PROGRAM AND BRING ON STRANGE RESULTS WHICH COULD BE DIFFICULT TO CORRECT FOR THE NOVICE PROGRAMMER. AS BOB MENTIONED USE THE ABREVIATED COMMANDS WITH CAUTION.

DISPLAY INVERSE CHARACTER SET
BY BOB KITCH

WHEN INVERSE CHARACTERS ARE SENT TO A DOT MATRIX PRINTER THE PRINTER SHIFTS TO GRAPHICS MODE AND REQUIRES A ROUTINE TO SUPPLY THE APPROPRIATE SHAPES TO THE HEAD. (NORMAL CHARACTERS ARE HELD IN THE PRINTERS ROM). IN THE VZ COMPUTER A TABLE OF SHAPES IS LOCATED AT 3B94H TO 3CD3H (15252 TO 15571 IN ROM.

THERE ARE 64 CHARACTERS, EACH USING 5 BYTES TO DEFINE THEIR GRAPHIC SHAPE. THE SHAPES MAY BE DECODED AND OUTPUT TO THE SCREEN AS IS DONE IN THIS PROGRAM. NOTE THERE ARE ERRORS IN THE ROM. THE 5 BYTES DEFINE A 5 BY 8 DOT MATRIX WHICH IS THE SHAPE OF THE CHARACTER, WHICH INCIDENTLY ARE NOT ORDERED ACCORDING TO THE ASCII CODE.

WHEN INVERSE CHARACTERS ARE SENT TO A DOT MATRIX PRINTER THE PRINTER SHIFTS TO GRAPHICS MODE AND REQUIRES A ROUTINE TO SUPPLY THE APPROPRIATE SHAPES TO THE HEAD. (NORMAL CHARACTERS ARE HELD IN THE PRINTERS ROM). IN THE VZ COMPUTER A TABLE OF SHAPES IS LOCATED AT 3B94H TO 3CD3H (15252 TO 15571 IN ROM.

THERE ARE 64 CHARACTERS, EACH USING 5 BYTES TO DEFINE THEIR GRAPHIC SHAPE. THE SHAPES MAY BE DECODED AND OUTPUT TO THE SCREEN AS IS DONE IN THIS PROGRAM. NOTE THERE ARE ERRORS IN THE ROM. THE 5 BYTES DEFINE A 5 BY 8 DOT MATRIX WHICH IS THE SHAPE OF THE CHARACTER, WHICH INCIDENTLY ARE NOT ORDERED ACCORDING TO THE ASCII CODE.

THE FIRST BYTE DEFINES THE LEFT HAND EDGE OF THE CHARACTER WHICH IS THE FIRST PRINTED DURING A PASS OF THE PRINTER HEAD. IN TANDY PRINTERS THE MSB IS THE LOWERMOST PIN OF THE HEAD AND LSB IS THE UPPERMOST PIN. THE PINS ON EPSON PRINTER HEADS ARE ARRANGED IN THE OPPOSITE SENSE. THIS REQUIRES THAT THE BITS IN EACH BYTE BE REVERSED.

```
010 '**************************************************************
020 '***        DISPLAY INVERSE CHARACTER  SET IN ROM          ***
030 '***            AS USED BY DOT MATRIX PRINTER              ***
040 '***              BY R. B. KITCH   27/9/86                 ***
050 '**************************************************************
090 :
100 DIM MK%(7) :'***VECTOR OF BIT MASK VALUES - POWERS OF 2
110 DIM BT%(7) :'***VECTOR OF DECODED BITS FROM ROM VALUE.
120 :
130 '***FILL MASK VECTOR WITH POWERS OF 2 FOR DECODING.
140 FOR I%=0 TO 7 :MK%(I%)=2^I% :NEXT I%
150 :
160 '***INITIALIZE PARAMETERS - MAY BE CHANGED TO VARY SCREEN.
170 CC%=4     :'***CHARACTER COLOUR.  (1-4)
180 BC%=2     :'***BACKGROUND COLOUR. (1-4)
190 CS%=0     :'***COLOUR SET. (0-1)
200 CW%=3     :'***COLUMN WIDTH BETWEEN CHARACTERS.
210 SP%=16    :'***ROW SPACING FOR CHARACTERS.
220 HS%=0     :'***STARTING HORIZONTAL POSITION ON HI-RES SCREEN.
230 VP%=3     :'***STARTING VERTICAL POSITION ON HI-RES SCREEN.
240 HM%=127   :'***MAXIMUM HORIZONTAL POSITION. (0-127)
250 :
260 '***SET UP MAIN LOOP TO STEP THROUGH ROM FROM 3B94H-3CD3.
270 BK%=0                    :'***BYTE COUNTER FOR EACH CHARACTER.
280 HP%=HS%                  :'***SET HORIZONTAL POSITION TO START
290 MODE(1) :COLOR,CS%       :'***SET HI-RES SCREEN AND COLOR SET.
300 FOR AD%=15252 TO 15571   :'***ROM ADDRESSES FOR SHAPE TABLE.
310   DV%=PEEK(AD%)          :'***DECIMAL VALUE IN ROM.
320 :
330 '***DECODE THE INDIVIDUAL BITS OF DV% AND STORE IN BT%().
340 '***THE MASK VALUES IN MK%() ARE "ANDED" WITH THE VALUE.
350 '***THE RESULT STORED IN BT%() IS THE "COLOUR" OF THE BIT.
360   FOR I%= 0 TO 7         :'***PROCEED FROM LSB TO MSB.
370     IF DV% AND MK%(I%) THEN BT%(I%)=BC% ELSE BT%(I%)=CC%
380   NEXT I%
390 :
400 '***CHECK THAT THERE IS ENOUGH ROOM TO PLOT CHARACTER.
410   IF BK%=0 AND HM%-HP%<4 THEN HP%=HS% :VP%=VP%+SP%:'*NEW ROW
420   BK%=BK%+1              :'***INCREMENT BYTE COUNTER.
430 :
440 '***OUTPUT BYTE TO SCREEN.
450   FOR I%=0 TO 7
```

```
460     COLOR BT%(I%)       :'***SET COLOUR OF BIT.
470     SET(HP%,VP%+I%)     :'***PLOT BIT.
480   NEXT I%
490 :
500 '***PREPARE FOR NEXT BYTE.
510   HP%=HP%+1              :'***INCREMENT HORIZONTAL POSITION.
520   IF BK%=5 THEN BK%=0 :HP%=HP%+CW%      :'***NEW CHARACTER.
530 NEXT AD%
590 GOTO 590
```

EDITOR'S COMMENTS:

THE VZ AS SOLD HAS A TANDY PRINTER COMPATIBLE BASIC ROM. FOR THOSE PERSONS WHO HAVE AN EPSON PRINTER COMPATIBLE EPROM THE ABOVE PROGRAM WILL PRINT THE INVERSE CHARACTERS UPSIDE DOWN. TWO MINOR CHANGES ARE REQUIRED TO FIX THE PROBLEM:

```
OLD LINE: 230 VP%=3          OLD LINE: 470 SET(HP%,VP%+I%)
NEW LINE: 230 VP%=11         NEW LINE: 470 SET(HP%,VP%-I%)
```

DISK USERS: ENTER THE FOLLOWING LINES FOR EXTRA FUNCTIONS.

```
590 GOTO 820
600 A$=INKEY$:A$=INKEY$:IF A$=""THEN 600
610 :
620 IF A$="U"THEN 700:'          ;GOES TO HI-RES SCREEN UPDATE
630 IF A$="S"THEN 710:'          ;GOES TO HI-RES SCREEN SAVE
640 IF A$="Q"THEN CLS:END:'      ;ENDS THIS PROGRAM
650 IF A$="^"THEN 800:'          ;GOES TO PROGRAM UPDATE ROUTINE
660 IF A$="&"THEN 810:'          ;GOES TO PROGRAM SAVE ROUTINE
670 GOTO 600
690 :
700 ERA"INV-ROM":'               ;ERASES HI-RES SCREEN FROM DISK
710 BSAVE"INV-ROM",7000,77FF:'   ;SAVES HI-RES SCREEN TO DISK
720 GOTO 820
790 :
800 :ERA"INV.ROM":'              ;ERASES THIS PROGRAM FROM DISK
810 SAVE"INV.ROM":'              ;SAVES THIS PROGRAM TO DISK
820 SOUND 25,1:GOTO 600
```

TAPE USERS: ENTER ONLY THE FOLLOWING LINES AFTER LINE 610.

```
620 IF A$="S"THEN 800:'          ;GOES TO PROGRAM SAVE ROUTINE
630 IF A$="Q"THEN CLS:END:'      ;ENDS THIS PROGRAM
640 GOTO 600
690 :
800 CSAVE"INVERSE ROM":'         ;SAVES THIS PROGRAM TO TAPE
820 SOUND 25,1:GOTO 600
```

ON THE NEXT PAGE YOU'LL SEE TWO HI-RES SCREEN PRINTOUTS PRODUCED BY THIS PROGRAM. THE LEFT ONE IS OF A STANDARD VZ ROM WHILE THE RIGHT ONE IS OF A MODIFIED VZ ROM (EPROM). IF YOU STUDY THEM CAREFULLY YOU'LL NOTE THE ERRORS IN THE LEFT ONE THAT BOB MENTIONED. ANOTHER ADVANTAGE OF THE MODIFIED EPROM IS THAT IT WILL PRINTOUT TO AN EPSON COMPATIBLE PRINTER ALL THE VZ'S INVERSE AND GRAPHIC CHARACTERS.

MODIFIED VZ ROM (EPROM):

BESIDES ALLOWING LLISTING AND LPRINTING OF ALL INVERSE AND GRAPHIC CHARACTERS THE EPROM ALSO PROVIDES ALL THE EXTENDED BASIC FUNCTIONS HOBBLED IN THE ORIGINAL VZ ROM WHICH MEANS IT IS NO LONGER REQUIRED TO LOAD ANY EXTENDED BASICS.

INVERSE CHARACTER SET CONT.


```
@ABCDEFGHIJKLMNO   @ABCDEFGHIJKLMNO
PQRSTUVWXYZ[\]↑←   PQRSTUVWXYZ[\]↑←
 !"#$%&'()*+,-./    !"#$%&'()*+,-./
0123456789:;<=>?   0123456789:;<=>?
```

BASIC & SOURCE CODE TO WP FILE CONVERTER BY DAVE MITCHELL

THIS UTILITY WHICH SOME OF YOU MAY KNOW AS BSTWP.F (BASIC AND SOURCE CODE TO WORD PROCESSOR FILE CONVERTER) IS USED IN CONJUNCTION WITH DAVE MITCHELL'S PATCH3.3 (DISK VERSION OF E & F WORD PROCESSOR). IT CAN ALSO BE USED WITH RUSSELL HARRISON'S VERSION OF DISK WORD PROCESSOR (WORDPRO).

AS THE NAME SUGGESTS THIS UTILITY CONVERTS BASIC PROGRAMS AND EDITOR ASSEMBLER ('S' FILE TYPE ONLY) SOURCE CODE FILES INTO WORD PROCESSOR FILES WHICH CAN BE LOADED INTO EITHER DISK WORD PROCESSOR. CONVERTING COULD'NT BE SIMPLER, IT'S JUST A MATTER OF ENTERING THE BASIC OR SOURCE CODE FILENAME AND ONCE LOADED ENTERING THE WP FILENAME. PRESSING RETURN WITHOUT FILENAME WILL PRESENT USER WITH DISK DIRECTORY IN 3 COLUMNS.

CONVERTING IS VERY FAST, DO'NT BLINK OR YOU MAY MISS IT. THE WP FILES ARE SAVED TO DISK WITH A 'W' FILETYPE WITH AN END ADDRESS OF D000H. BOTH BASIC AND SOURCE CODE FILES ARE PADDED OUT TO 5 DIGITS WITH LEADING ZEROES WHICH YOU MAY HAVE COME ACROSS IN THE JOURNAL ON NUMEROUS OCCASIONS.

AS YOU ENTER THE SOURCE CODE SAVE IT EVERY SO OFTEN SO YOU WON'T LOSE ALL YOUR TYPING. SAVE THE SOURCE CODE AS CONV.SRC AND THE OBJECT CODE AS CONVERT.

WARNING: YOU SHOULD BE AWARE THAT ANY GRAPHIC CHARACTERS IN YOUR CONVERTED BASIC PROGRAM WILL PRINT OUT AS GARBAGE AND IS A LIMITATION OF THE WORD PROCESSOR.

ANYONE WHO HAS TRIED THE PRINTER FUNCTION OF THEIR EDITOR ASSEMBLER WILL FIND THIS UTILITY VERY HANDY FOR PRINTING OUT SOURCE CODE FILES AS IT CAN TAKE ADVANTAGE OF ALL THE PRINTER FUNCTIONS OF PATCH3.3 OR WORDPRO HAS TO OFFER.

CONVERT VERSION 3.1A:

BRIAN GREEVE OF WA HAS MODIFIED CONVERT BY ADDING EXTRA FUNCTIONS MAKING IT VERY VERSATILE INDEED. THANKS BRIAN. CONV3.1A WILL CONVERT ALL SOURCE CODE FILETYPES, EG: S, A & W AS WELL AS QUICK WRITE 'F' WORD PROCESSOR FILES. OTHER OPTIONS ARE NUMBER OF DIGITS SELECT, 1-5 AND D000 OR FE00 WP FILE END ADDRESS FOR PATCH3.3 OR WORDPRO.

44-11 BASIC & SOURCE CONV. CONT.

```
;BASIC AND SOURCE CODE
;TO WORD PROCESSOR FILE.
;BY D.MITCHELL - 07B00H
        LD      SP,7AFFH
        PUSH    IY
        LD      HL,(78B1H)
        LD      BC,END-HO
        PUSH    BC
        XOR     A
        SBC     HL,BC
        LD      (78B1H),HL
        INC     HL
        PUSH    HL
        XOR     A
        LD      BC,33H
        SBC     HL,BC
        LD      (78A0H),HL
        DEC     HL
        LD      (78E8H),HL
        LD      DE,HO
        POP     HL
        PUSH    HL
        XOR     A
        SBC     HL,DE
        EX      DE,HL
        LD      B,33
        LD      IX,LT
RC      LD      L,(IX+00)
        LD      H,(IX+01)
        PUSH    BC
        PUSH    HL
        PUSH    HL
        POP     BC
        POP     IY
        LD      L,(IY+00)
        LD      H,(IY+01)
        ADD     HL,DE
        LD      A,L
        LD      (BC),A
        INC     BC
        LD      A,H
        LD      (BC),A
        INC     IX
        INC     IX
        POP     BC
        DJNZ    RC
        POP     DE
        LD      (7200H),DE
        LD      HL,HO
        POP     BC
        LDIR
        POP     IY
L1      LD      HL,ERR
        LD      (79ADH),HL
        LD      A,0C3H
        LD      (79ACH),A
        LD      HL,7AE9H
        LD      (78A4H),HL
        CALL    1B4DH
        LD      HL,(7200H)
        JP      (HL)
LT      DEFW    L1+1
        DEFW    HO+1
        DEFW    E1+1
        DEFW    L2+1
        DEFW    L3+1
        DEFW    L4+1
        DEFW    L5+1
        DEFW    L6+1
        DEFW    A0+1
        DEFW    L7+1
        DEFW    L8+1
        DEFW    L9+1
        DEFW    L10+1
        DEFW    L11+1
        DEFW    B0+1
        DEFW    L12+1
        DEFW    L13+1
        DEFW    L14+1
        DEFW    L15+1
        DEFW    L16+2
        DEFW    L17+2
        DEFW    L18+1
        DEFW    L18A+1
        DEFW    L19+1
        DEFW    E2+1
        DEFW    L20+1
        DEFW    L21+1
        DEFW    L22+1
        DEFW    IPUT+1
        DEFW    L23+1
        DEFW    ERR+2
        DEFW    M1+1
        DEFW    M2+1
HO      LD      HL,MES0
        CALL    2B75H
L2      LD      HL,M1
L3      LD      (ADD),HL
M1      LD      HL,MES1
        CALL    2B75H
E1      CALL    IPUT
L4      CALL    DLY
        CALL    402CH
        CP      2
        JR      Z,S2
S1      OR      A
        JP      NZ,400EH
        LD      A,0DH
        JR      S1
S2      LD      A,(IY+10)
        CP      "T"
        JR      Z,S3
        CP      "S"
        JR      Z,S3
        LD      A,0CH
        JR      S1
S3      LD      (IY+9),A
        CALL    4041H
        OR      A
        JP      NZ,400EH
        CALL    400BH
        LD      A,(IY+9)
        CP      "T"
        JR      NZ,B0A
```

```
        LD    DE,4
        LD    HL,(78F9H)
        ADD   HL,DE
L5      LD    (BUF1),HL
L6      LD    (BUF2),HL
        LD    HL,(78A4H)
A0      CALL  EXIT
        LD    A,(HL)
        INC   HL
        LD    (7921H),A
        LD    A,(HL)
        INC   HL
        LD    (7922H),A
        PUSH  BC
        PUSH  HL
L7      CALL  BTA
L8      LD    (BUF1),HL
        LD    (78A7H),HL
        POP   HL
        CALL  2B7EH
        LD    HL,(78A7H)
A1      LD    A,(HL)
        OR    A
        JR    Z,A3
        CP    40H
        JR    C,A2
        CP    5BH
        JR    NC,A2
        ADD   A,20H
        LD    (HL),A
A2      INC   HL
        JR    A1
A3      LD    (HL),0DH
        INC   HL
        LD    (HL),0DH
        INC   HL
L9      LD    (BUF1),HL
        LD    HL,79E8H
        LD    (78A7H),HL
        POP   HL
        JR    A0
B0A     LD    HL,(78F9H)
        LD    (HL),0
        INC   HL
        LD    (HL),0
        LD    HL,8000H
L10     LD    (BUF1),HL
L11     LD    (BUF2),HL
        LD    HL,0A280H
B0      CALL  EXIT
        INC   HL
        LD    (7921H),BC
        PUSH  HL
L12     CALL  BTA
        EX    DE,HL
        POP   HL
        LD    A,(HL)
        CP    7BH
        JR    Z,B1
        CP    7AH
        JR    Z,B1
        CP    6AH
        JR    Z,B1
        LD    B,5
L13     CALL  C0
        LD    B,5
L14     CALL  C0
        LD    B,12H
L15     CALL  C0
        JR    B4
B1      LD    A,(HL)
        CP    0FFH
        JR    Z,B4
        CP    5BH
        JR    NC,B3
        ADD   A,20H
B2      LD    (DE),A
        INC   DE
        INC   HL
        JR    B1
B3      RES   6,A
        JR    B2
B4      LD    A,0DH
        LD    (DE),A
        INC   DE
        LD    (DE),A
        INC   DE
L16     LD    (BUF1),DE
        INC   HL
        JR    B0
C0      LD    A,(HL)
        CP    60H
        JR    Z,C2
        CP    0FFH
        RET   Z
        CP    5BH
        JR    NC,C1
        ADD   A,20H
        JR    C3
C1      RES   6,A
        JR    C3
C2      LD    A,20H
        DEC   HL
C3      LD    (DE),A
        INC   DE
        INC   HL
        DEC   B
        JR    NZ,C0
        INC   HL
        RET
EXIT    LD    C,(HL)
        INC   HL
        LD    B,(HL)
        INC   HL
        LD    A,B
        OR    C
        RET   NZ
L17     LD    DE,(BUF2)
L18     LD    HL,(BUF1)
        LD    (HL),0DH
        PUSH  HL
        OR    A
        SBC   HL,DE
        JR    Z,EI
```

44-13 BASIC & SOURCE CONV. CONT.

```
249         PUSH HL
250         POP  BC
251         INC  BC
252         LD   HL,0CFFFH
253         EX   DE,HL
254         POP  HL
255         LDDR
256         INC  DE
257         LD   (78A4H),DE
258         LD   HL,0D000H
259         LD   (HL),0
260         LD   (78F9H),HL
261 L18A    LD   HL,M2
262 L19     LD   (ADD),HL
263 M2      LD   HL,MES2
264         CALL 2B75H
265 E2      CALL IPUT
266         LD   (IY+9),"W"
267 L20     CALL DLY
268         IN   A,(13H)
269         BIT  7,A
270         JR   Z,S4
271         LD   A,4
272         JP   400EH
273 S4      CALL 4044H
274         OR   A
275         JP   NZ,400EH
276         CALL 400BH
277 EI      EI
278 L21     JP   HO
279 BTA     LD   BC,0606H
280 L22     LD   HL,(BUF1)
281         CALL 132FH
282         DEC  HL
283         LD   (HL),20H
284         INC  HL
285         RET
286 DLY     DI
287         CALL 4008H
288         LD   BC,0032H
289         CALL 4038H
290         RET
291 BR      PUSH IY
292         POP  HL
293         DEC  HL
294         LD   (78B1H),HL
295         LD   BC,65485
296         ADD  HL,BC
297         LD   (78A0H),HL
298         LD   HL,7AE9H
299         LD   (78A4H),HL
300         LD   A,0C9H
301         LD   (79ACH),A
302         CALL 1B4AH
303         JP   1A19H
304 IPUT    LD   HL,MES3
305         CALL 2B75H
306         CALL 053AH
307         JR   C,BR
308         LD   A,(DE)
309         CP   20H
310         JR   Z,DIR
311         LD   A,20H
312         LD   (DE),A
313         INC  HL
314         PUSH IY
315         POP  DE
316         INC  DE
317         LD   BC,8
318         LDIR
319         RET
320 DIR     CALL 01C9H
321         XOR  A
322         LD   (IY+11H),A
323         LD   (IY+12H),A
324         LD   (7AAEH),A
325 L23     CALL DLY
326 D1      DI
327         CALL 4035H
328         CP   11H
329         JR   Z,D7
330         OR   A
331         JP   NZ,400EH
332         LD   L,(IY+31H)
333         LD   H,(IY+32H)
334         LD   B,8
335 D2      LD   A,(HL)
336         OR   A
337         JR   Z,D7
338         PUSH BC
339         CP   1
340         JR   NZ,D3
341         LD   DE,16
342         ADD  HL,DE
343         JR   D6
344 D3      LD   B,10
345         EI
346 D4      LD   A,(HL)
347         INC  HL
348         CALL 033AH
349         DJNZ D4
350         LD   A,(7AAEH)
351         ADD  A,10
352         AND  20H
353         JR   NZ,D5A
354         LD   A,20H
355         CALL 033AH
356         JR   D5
357 D5A     LD   A,0DH
358         CALL 033AH
359         LD   A,1BH
360         CALL 033AH
361 D5      LD   DE,6
362         ADD  HL,DE
363 D6      POP  BC
364         DJNZ D2
365         INC  (IY+11H)
366         LD   A,(IY+11H)
367         CP   0FH
368         JR   NZ,D1
369 D7      CALL 400BH
370         EI
371         POP  BC
372 ERR     LD   BC,(ADD)
```

```
373        PUSH BC
374        RET
375 ADD    DEFW 0
376 BUF1   DEFW 0
377 BUF2   DEFW 0
378 MES0   DEFB 1FH
379 *  THIS PROGRAM WILL MAKE WO
380 *RD*
381        DEFB 0DH
382 * PROCESSOR FILES FROM*
383 * BASIC AND  EDITOR ASSEMBL*
384 *ER SOURCE CODES*
385        DEFB 0DH
386 *      WRITTEN BY D.MITCHELL*
387        DEFB 0DH
388        NOP
389 MES1   DEFB 0DH
390 * ENTER B/S *
391        NOP
392 MES2   DEFB 0DH
393 * ENTER WP. *
394        NOP
395 MES3   EQU  $
396 *FILENAME :*
397        NOP
398 END    NOP
```

VZ NUMBERS FREQUENCY

THIS PROGRAM IS DESIGNED TO KEEP TRACK OF WINNING NUMBERS IN POOLS AND GO LOTTO. THERE IS A CHOICE OF PRIME NUMBERS ONLY OR CAN INCLUDE SUPPLEMENTARY NUMBERS AS WELL. WITH SOME MINOR MODIFICATIONS THIS PROGRAM COULD BE ALTERED TO KEEP TRACK OF OTHER NUMBER FREQUENCIES.

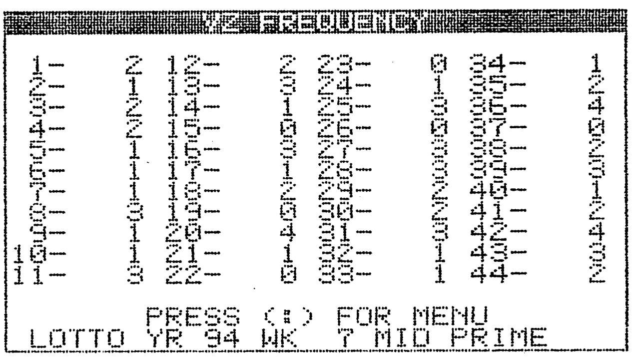
THE LEFT PRINTOUT BELOW SHOWS GO LOTTO FREQUENCY NUMBERS FOR THE YEAR OF 1994 UP TO THURSDAY, WEEK 7 WHILE THE ONE ON THE RIGHT IS FOR THE SAME TIME PERIOD, BUT INCLUDES SUPPLEMENTARY NUMBERS AS WELL.

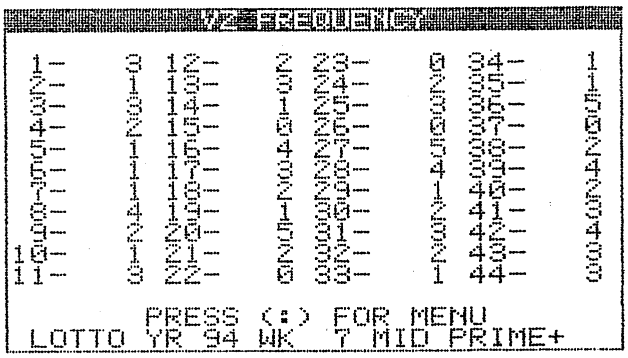
THERE ARE TWO INPUT CHOICES:

1) INSERT - THIS ALLOWS YOU TO INSERT NUMBERS AS THEY ARE DRAWN AND/OR ADD TO EXISTING NUMBERS.
2) EDIT - IF YOU HAVE BEEN KEEPING TABS OF NUMBERS USING PENCIL AND PAPER THEN THIS OPTION WILL ALLOW YOU TO ENTER NUMBERS IN THE RANGE FROM 0 TO 9999 MAXIMUM.

ALSO TO THE BEST OF MY KNOWLEDGE THIS IS THE FIRST TIME THAT A 4 BYTE MULTI USR CALL ROUTINE HAS BEEN USED IN A PUBLISHED PROGRAM WHICH WAS DESCRIBED IN ISSUE # 31, PAGE 14. IT SIMPLIFIES THE CALLING OF M/C ROUTINES WHICH ARE:

1) VZ SOUND BEEPER.
2) M/C CLS (CLEAR PART SCREEN - 14 LINES.)
3) M/C DIRECTORY ROUTINE WHICH WILL BE PUBLISHED IN NEXT ISSUE.





```
10 GOTO1000
12 ERA"---------":RETURN
14 REN"---------","--------":RETURN
16 OPEN"---------",0
18 IN#"---------",YR,WK,R,L$,P$:RETURN
20 FORI=1TOR
22 IN#"---------",C(I):NEXT
24 CLOSE"---------":RETURN
26 SAVE"---------":RETURN
34 :
36 '"[illegible]
38 :
40 FOR O=0 TO 7:POKE AD+O,ASC(MID$(N$+"        ",O+1,1))
42 NEXT:RETURN
44 :
46 FOR O=0 TO 2:POKE AD+O,ASC(MID$(X$,O+1,1)):NEXT:RETURN
48 :
50 CLS:POKE 30777,1:PRINT"[illegible]";
52 IF M>0 THEN RETURN:'32 INVERSE SPACES ^^
54 PRINT@9,FR$:RETURN
58 :
60 PRINT@275,"N":PRINT@258,"ARE YOU SURE   ";:RETURN
62 :
64 PRINT@487,M$;:RETURN
66 :
68 PRINT@226,"SELECTED DRIVE >";D:RETURN
78 :
80 ND=0:FOR I=1 TO R:IF C(I)>0 THEN ND=1:I=R:'"[illegible]
82 NEXT:IF ND=1 THEN RETURN
84 PRINT@393,"[illegible]":SOUND 5,6:ND=0:RETURN
88 :
90 IF R=44 THEN PL$="LOTTO" ELSE PL$="POOLS"
91 PRINT@481,PL$;:PRINT" YR    WK";
92 PRINT@490,USING"##";YR;
93 PRINT@496,USING"##";WK;
94 PRINT@499,USING"% %";L$;
95 PRINT@503,P$;:RETURN
98 :
100 AD=31535:GOSUB 40:'POKE LINE 16 FILENAME
106 AD=31555:GOSUB 40:'POKE LINE 18 FILENAME
108 AD=31600:GOSUB 40:'POKE LINE 22 FILENAME
110 AD=31627:GOSUB 40:'POKE LINE 24 FILENAME
112 AD=31551:GOSUB 46:'POKE LINE 18 IN#/PR#
114 AD=31596:GOSUB 46:'POKE LINE 22 IN#/PR#
116 POKE 31545,X      :'POKE LINE 16 0 OR 1
118 :
120 IF U=2 THEN AD=31483:GOSUB 40:GOSUB 12:'POKE LINE 12 F'NAME
122 GOSUB 16:GOSUB 90:GOSUB 20
124 IF U=0 THEN 500 ELSE 400
148 :
150 M=0:GOSUB 50:GOSUB 90:'"[illegible]
152 PRINT@247,YR:PRINT@226,"ENTER YEAR  (94-99) ";
154 INPUT YR:IF YR=0 THEN 300
156 IF YR<94 OR YR>99 THEN 152
158 :
160 PRINT@279,WK:PRINT@258,"ENTER WEEK  (00-52) ";:INPUT WK
162 IF WK=0 THEN 300
164 IF WK>52 THEN 160
168 :
170 PRINT@312,L$:PRINT@290,"ENTER (MON MID SAT) ";:INPUT L$
172 IF L$="" THEN 300
174 L$=LEFT$(L$,3):GOTO 300
```

```
198 :
200 M=0:GOSUB 50:GOSUB 90:'"[illegible]
202 :
204 IF R=44 THEN PRINT@243,"L" ELSE PRINT@243,"P"
206 PRINT@226,"LOTTO OR POOLS ";:INPUT Y$
208 :
210 IF Y$="L"THEN R=44:P=8:PRINT@481,"[illegible]";:GOTO 216
212 IF Y$="P"THEN R=38:P=7:PRINT@481,"[illegible]";:ELSE 204
214 :
216 PRINT@275,"N":PRINT@258,"PRIME #'S ONLY ";:INPUT P$
218 :
220 IF P$="Y"THEN P$="PRIME ":P=6:GOTO 230
222 IF P$="N"THEN P$="PRIME+":GOTO 230 ELSE 216
224 :
230 X=USR(31348):'CLEAR PART SCREEN
232 L=98:FOR A=1 TO P
234 :
236 IF A<7 THEN PRINT@L,"ENTER WINNING #";A;"-> ";
238 IF A=7 THEN L=322
240 IF A>6 THEN PRINT@L,"SUPPLEMENTARY NUMBER ";
242 :
244 INPUT N(A)
246 IF N(A)=0 THEN A=P:GOTO 300
248 IF A<7 THEN IF N(A)>R THEN 236
250 IF A>6 THEN IF N(A)>R THEN 240
252 T=0:IF A<>1 THEN FOR I=1 TO A-1:T=T-(N(I)=N(A)):NEXT I
254 IF (T) THEN 236
256 L=L+32:NEXT
258 :
260 PRINT@441,"Y":PRINT@418,"ARE NUMBERS CORRECT  ";
262 :
264 INPUT Y$:IF Y$="Y"THEN 270
266 IF Y$="N"THEN 232 ELSE 260
268 :
270 FOR I=1 TO P:'"[illegible]
272 C(N(I))=C(N(I))+1:NEXT
274 :
276 PRINT@441,"Y":PRINT@418,"ENTER MORE NUMBERS   ";:INPUT Y$
278 :
280 IF Y$="Y"THEN 230
282 IF Y$="N"THEN 500 ELSE 276
298 :
300 M=0:GOSUB 50:'"[illegible]
302 :
304 PRINT@073,"I) INSERT  DATA"
306 PRINT@105,"E) EDIT    DATA"
308 PRINT@137,"D) DISPLAY DATA"
310 :
312 PRINT@201,"P) PRINT   DATA"
314 PRINT@233,"C) CLEAR   DATA"
316 PRINT@265,"Y) ENTER  YR/WK"
318 :
320 PRINT@329,"Q) QUIT & PARK"
324 PRINT@361,";) SUB  MENU"
326 PRINT@393,":) DISK MENU   "
342 :
344 GOSUB 90:SOUND 25,1
346 POKE 30777,35
348 :
350 C$=INKEY$:C$=INKEY$:IF C$=""THEN 350 ELSE X=USR(13392)
352 :
354 IF C$="I"THEN 200
```

```
356 IF C$="E"THEN 900
358 IF C$="D"THEN 500
360 :
362 IF C$="P"THEN 550
364 IF C$="C"THEN RUN
366 IF C$="Y"THEN 150
368 :
370 IF C$="Q"THEN 680
372 IF C$=";"THEN 960
374 IF C$=":"THEN 400
378 :
396 GOTO 350
398 :
400 M=0:GOSUB 50:'"[illegible]
402 :
404 PRINT@073,"L) LOAD   DATA"
406 PRINT@105,"S) SAVE   DATA"
408 PRINT@137,"U) UPDATE DATA"
410 :
412 PRINT@201,"E) ERASE  FILE"
414 PRINT@233,"R) RENAME FILE"
416 PRINT@265,"I) INIT DISK"
418 :
420 PRINT@329,"D) DIRECTORY"
422 PRINT@361,"1) DRIVE 1/2"
424 PRINT@393,":) MAIN MENU   "
428 :
430 DRIVE D:PRINT@361,USING"#";DR
442 :
444 GOSUB 90:SOUND 25,1
446 POKE 30777,35
448 :
450 D$=INKEY$:D$=INKEY$:IF D$=""THEN 450 ELSE X=USR(13392)
452 :
454 IF D$="L"THEN U=0:GOTO 650
456 IF D$="S"THEN U=1:GOTO 650
458 IF D$="U"THEN U=2:GOTO 650
459 :
460 IF D$="E"THEN 600
462 IF D$="R"THEN 620
463 :
464 IF D$="D"THEN 850
466 IF D$="I"THEN 780
470 IF D$="1"THEN D=1:DR$="[illegible]":GOTO 430
472 IF D$="2"THEN D=2:DR$="[illegible]":GOTO 430
474 :
476 IF D$=":"THEN 300
494 :
496 GOTO 450
498 :
500 GOSUB 80:IF ND=0 THEN 326:'"[illegible]
502 M=0:GOSUB 50:GOSUB 90
508 :
510 E$="##":F$="#### ":PRINT@64,"";
518 :
520 IF R=38 THEN X1=10:Y1=30:ST=10
522 IF R=44 THEN X1=11:Y1=33:ST=11
524 FOR X=1 TO X1:FOR Y=0 TO Y1 STEP ST
526 IF R=38 THEN IF X+Y>38 THEN PRINT:GOTO 534
528 :
530 PRINT USING E$;X+Y;:PRINT"-";
532 PRINT USING F$;C(X+Y);
```

```
534 NEXT Y:NEXT X
536 E$="":F$=""
542 :
544 PRINT@455,M$
546 IF INKEY$=":"THEN 300 ELSE 546
548 :
550 GOSUB 80:IF ND=0 THEN 326:'"[illegible]
552 M=1:GOSUB 50:PRINT@11,"[illegible]":GOSUB 90
554 :
556 GOSUB 60:INPUT Y$
558 IF Y$="N"THEN 300
560 IF Y$="Y"THEN 564 ELSE 556
562 :
564 E$="##":F$="**## "
566 LPRINT TAB(10)"VZ ";PL$;" YR ";USING E$;Y;
568 LPRINT" WK ";USING E$;W;:LPRINT" ";USING"%  %";L$;
570 LPRINT USING"%     %";P$:LPRINT
572 :
574 IF R=38 THEN X1=10:Y1=30:ST=10
576 IF R=44 THEN X1=11:Y1=33:ST=11
578 FOR X=1 TO X1:FOR Y=0 TO Y1 STEP ST
580 IF R=38 THEN IF X+Y>38 THEN 590
584 :
586 LPRINTTAB(10)USING E$;X+Y;:LPRINT"-";
588 LPRINT USING F$;C(X+Y);
590 NEXT Y:LPRINT
594 NEXT X:LPRINT
596 E$="":F$="":GOTO 300
598 :
600 M=1:GOSUB 50:PRINT@10,"[illegible]";:GOSUB 68:PRINT@258,EF$;
602 :
604 SOUND 30,1:INPUT N$
606 IF N$=""THEN 400
608 IF LEN(N$)>8 THEN 600
610 AD=31483:GOSUB 40:GOSUB 12
612 GOTO 400
618 :
620 M=1:GOSUB 50:PRINT@10,"[illegible]";:GOSUB 68
622 :
624 PRINT@258,"OLD FILENAME > ";:SOUND 30,1:INPUT N$
626 IF N$=""THEN 400
628 IF LEN(N$)>8 THEN 620
630 AD=31503:GOSUB 40
632 :
634 PRINT@290,"NEW FILENAME > ";:SOUND 30,1:INPUT N$
636 IF N$=""THEN 400
638 IF LEN(N$)>8 THEN 634
640 AD=31514:GOSUB 40
642 GOSUB 14:GOTO 400
648 :
650 IF U>0 THEN GOSUB 80:IF ND=0 THEN 424:'"[illegible]
652 M=1:GOSUB 50
654 IF U=0 THEN PRINT@11,"[illegible]";
656 IF U=1 THEN PRINT@11,"[illegible]";
658 IF U=2 THEN PRINT@10,"[illegible]";
660 PRINT"[illegible]":GOSUB 90:GOSUB 68
662 :
664 PRINT@275,N$:PRINT@258,EF$;
666 INPUT N$:IF N$=""THEN 400
668 IF LEN(N$)>8 THEN664
```

. . CONTINUED NEXT ISSUE . .

44-19 VZ-EPSON PRINTER PATCH V1.4

BY LARRY TAYLOR - PUBLIC DOMAIN

The VZ-EPSON PRINTER PATCH allows EPSON or EPSON compatible printers to LPRINT or LLIST all of the VZ's inverse and graphics characters.

These same characters can also be dumped to the printer from the LO-RES screen, when the COPY command is used. In MODE(1), use of the COPY command will dump the HI-RES screen to the printer. In addition, an extension of the COPY command has been provided. The new command is COPYA. If COPYA is used in MODE(0), a dump of the LO-RES screen will be made without the usual linefeeds, whilst in MODE(1), a larger than normal dump of the HI-RES screen will be printed. To cater to differences in line feeds between printers, both commands may be used in the form COPYn and COPYAn, where n is a value ranging from 4 to 8. Changing n allows selection of a line feed setting appppropriate to the printer being used. Should COPY or COPYA not be followed by a line feed value, then a default setting of 6 will be used.

Three additional commands, LTAB(n), PAGE(n) and FEED(n) have also been added. The LTAB(n) command is used to set a left margin for any output to the printer, whether from an LPRINT statement, during LLISTing of a BASIC program or when doing a screen dump. The value n may be in the range 0 to 63, indicating the number of spaces in from the left that printing will occur. A printout of the disk directory at the current LTAB(n) position can be obtained by first setting the output device code to printer, followed by the DIR command. eg. POKE30876,1:DIR

PAGE(n) sets the page length and n may be in the range 0 to 127. FEED(n) sets the number of linefeeds between the end of printed text and the end of the page. It can be in the range from 0 to 127, but unless it is less than the page length, it will be ignored. Setting a valid linefeed value, causes a form feed to occur automatically. If n is set to 0 then the automatic form feed is switched off. LTAB(n), PAGE(n), FEED(n), LLIST, LPRINT and both the COPY commands may be entered either directly from the keyboard, or used from within a program, but will only function correctly whilst the PRINTER PATCH is present.

The utility locates and protects itself according to available memory, allowing BASIC programs to be loaded and worked on normally. When LLISTing a BASIC program, the routine is also able to recognize all of the hidden commands, which may have been entered using an EXTENDED BASIC. Although not dependent on it, the routine can be used with Steve Olney's EXTENDED BASIC already resident in memory and will not interfere with it in any way. To use it in this way, first load the extended BASIC and then the PRINTER PATCH. Once the READY message appears you will have access to both the EXTENDED BASIC and the improved printer commands.

NEW PUBLIC DOMAIN SOFTWARE

DAVE MITCHELL

E & F WP PATCH 3.3
MENU & FILE COPIER
EXTENDED DOS V1.3
MONITOR 6000H
DISK FILER & CATALOGUER

E & F WP PATCH 3.4
MENU & FILE COPIER 6000H
EXTENDED DOS 6000H
DISK FILE COPIER 6000H

MARK HARWOOD

MARK 32 DATABASE
DISK EDITOR ASSEMBLER

MARK 64 DATABASE
VZ USER NEWSLETTERS - 22 ISSUES

LARRY TAYLOR

VZ EPSON PRINTER PATCH 1.4
COPYPRO 4: FILE COPIER/PROTECTOR
COPYPRO 5: TWO DRIVE DISK ONLY
DISK COPY 2: SINGLE DRIVE COPIER
FILE SEARCH 2 (DISK/TAPE CONTENTS)
DISK MENU (MEMORY RESIDENT)

ESCAPE RIVER (LO/HI-RES ADVENTURE)
FACTORY (EDUCATIONAL GAME)
TOWER OF HANOI (EDUCATIONAL GAME)
NAME THE TOWNS (EDUCATIONAL)
MICROSCOPE (EDUCATIONAL)
SNERTLE (EDUCATIONAL MATHS)

BRIAN GREEVE: MACHINE CODE DATA 2

H.V. VZ U.G.: HUNTER VALLEY VZ JOURNAL

DICK SMITH EL: IT COULD BE SAFELY SAID THAT IT IS PUBLIC DOMAIN ALSO AS FOR A COUPLE YEARS NOW THEY HAVE REFERED TO ME PERSONS WISHING TO AQUIRE VZ SOFTWARE, HARDWARE OR VARIUOS PUBLICATIONS.

NOTE: I HAVE MOST OF THE ABOVE PROGRAMS AND THEY CAN BE HAD JUST FOR THE PRICE OF COVERING COSTS LIKE POST, PACKING, DISKS, ETC.

CONTRIBUTIONS TO THE JOURNAL

IF YOU ARE THINKING OF CONTRIBUTING TO THE JOURNAL THE PREFERED FORMAT IS BASIC LISTINGS, WORD PROCESSOR OR SOURCE CODE FILES ON TAPE OR DISK. FILES FROM ALL EDITOR ASSEMBLERS AND WORD PROCESSORS CAN BE ACCEPTED. THE PREFERED WORD PROCESSORS ARE LESLIE MILBURN QUICK WRITE, RUSSELL HARRISON WORDPRO AND DAVE MITCHELL PATCH 3.3.

CLUB MEETINGS — ALL WELCOME

MEETINGS WILL BE APPROXIMATELY ONCE A MONTH, USSUALY ON THE FIRST FRIDAY. MEETING DATES WILL BE ADJUSTED TO ACCOMODATE MEMBERS ABILITY TO ATTEND. WHETHER YOU ARE A LOCAL MEMBER, INTRA OR INTER STATE VISITOR PLEASE CHECK WITH JOE LEON FIRST.

HUNTER VALLEY VZ JOURNAL SUBSCRIPTIONS

SUBSCRIPTION TO - AUST. - 3 ISSUES $11.00 - 6 ISSUES $21.00
H.V.VZ.JOURNAL - N. Z. - 3 ISSUES $13.00 - 6 ISSUES $26.00
PRICES INCLUDE POST & PACKING

FOR MORE INFORMATION ON MEETINGS AND SUBS CONTACT:
JOE LEON 35 TIGHES TCE TIGHES HILL 2297 (049) 611 474 AUSTRALIA

NOTE: WHEN WRITING TO H.V.VZ. USERS' GROUP FOR INFORMATION PLEASE ENCLOSE A S.S.A.E. OR NZ 2 INT. REPLY COUPONS.